

**IMPLEMENTACIÓN DEL DICCIONARIO UNADISTA Y DESARROLLO DE LA
SEGUNDA FASE DEL MÓDULO ASSISTAPP BAJO UNA SOLUCIÓN MÓVIL
QUE PERMITA LA GEORREFERENCIA DE USUARIOS Y LA INSCRIPCIÓN A
EVENTOS DEL CCAV PITALITO**

SEBASTIÁN GAMBOA OSORIO.

CC. 1083900153

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA UNAD.

INGENIERÍA DE SISTEMAS

PITALITO - HUILA.

2019

**IMPLEMENTACIÓN DEL DICCIONARIO UNADISTA Y DESARROLLO DE LA
SEGUNDA FASE DEL MÓDULO ASSISTAPP BAJO UNA SOLUCIÓN MÓVIL
QUE PERMITA LA GEORREFERENCIA DE USUARIOS Y LA INSCRIPCIÓN A
EVENTOS DEL CCAV PITALITO**

SEBASTIÁN GAMBOA OSORIO.

CC. 1083900153

INGENIERIA DE SISTEMAS

ASESOR:

JHON JAIRO PEREZ

INGENIERO DE SISTEMAS

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA UNAD.

INGENIERÍA DE SISTEMAS

PITALITO - HUILA.

2019

DEDICATORIA

Dedico este proyecto principalmente a Dios, por permitirme cumplir una de las metas que más anhelo en mi vida y en el transcurso del camino cruzarme con tanta gente maravillosa que ha aportado su granito de arena para convertirme en el profesional que soy.

A mi madre, por confiar siempre en mis capacidades y por siempre transformar mis metas en sus metas, para con su apoyo llegar a este momento tan importante de mi formación profesional.

A mi padre, porque a pesar de las dificultades siempre está dispuesto ayudarme y apoyarme en cada paso que doy.

A mi esposa, porque en ella encontré un motivo para salir adelante y superarme personal y profesionalmente.

A mis tutores en la UNAD, que gracias a su apoyo, persistencia y paciencia lograron inspirar en mí un gran sentido de responsabilidad y compromiso conmigo y mi entorno profesional.

AGRADECIMIENTOS

En estas líneas quiero agradecer a mis padres por el esfuerzo, dedicación, paciencia, por su confianza y por todo lo que me han dado a lo largo de mi carrera y de mi vida.

Quiero expresar mi gratitud a las personas encargadas de registro y control y a los involucrados en el desarrollo de la primera fase de la aplicación AssistApp, por confiar en mi capacidad y permitirme realizar el proceso investigativo.

También quiero agradecer a los tutores de la UNAD Pitalito porque siempre están dispuestos para apoyarme en cada proceso y en el transcurso del desarrollo del proyecto siempre tuvieron la mejor disposición para enseñarme y guiarme por el camino más conveniente para que mi desarrollo profesional sea el mejor.

De manera especial quiero expresar mi agradecimiento a mi tutor de proyecto, por haberme guiado, no solo en la elaboración de este trabajo de titulación, sino a lo largo de mi carrera universitaria y haberme brindado el apoyo para desarrollarme profesionalmente y seguir cultivando mis valores profesionales.

RESUMEN

La terminología, siglas y referencias que se usan en la UNAD son propias la mayoría no son de uso cotidiano, una experiencia nueva para cada estudiante que al ingreso debe familiarizarse aprender y entender de manera autónoma y ayudado por algunos talleres que la universidad ofrece.

Actualmente el manejo que tiene la Universitaria UNAD con respecto al registro de asistencia de los docentes y estudiantes que asisten a encuentros presenciales tales: conferencias, foros, capacitaciones, talleres entre otros es llevado de forma manual en planillas. Este procedimiento demanda tiempos largos y genera exceso de papel al momento de ser archivados, dicho método, además, en ocasiones no es confiable debido al margen de error al que se está expuesto, pues el control y gestión de la información se ve afectada por algunos factores como la manejabilidad del soporte físico (papel), dificultad para modificar el contenido y su vida útil puesto que su durabilidad va decreciendo con el tiempo, igualmente, puede presentarse desaparición o deterioro de la información lo cual conllevaría a su pérdida total.

Por otra parte la UNAD cuenta actualmente con una plataforma web llamada AssistApp que se usa para crear y confirmar asistencia de eventos que realiza la universidad como B-LEARNING, CIPAS, COMPONENTES PRÁCTICOS entre otros esta versión funciona perfectamente y es de utilidad para tutores y estudiantes ya que minimiza procesos y centraliza la información para un fácil manejo de la misma en los módulos de usuario se encuentran 2 opciones una para ver el calendario de actividades por día, semana o mes y otra opción para ver los eventos del día y en la parte administrativa se encuentra el módulo para crear, ver, modificar o eliminar eventos.

CONTENIDO

| | |
|--|------------------|
| <i>CAPITULO I. GENERALIDADES DEL PROYECTO.....</i> | <i>13</i> |
| 1.1 TÍTULO | 13 |
| 1.2 EL PROBLEMA | 13 |
| 1.2.1 DESCRIPCIÓN DEL PROBLEMA..... | 13 |
| 1.2.2 FORMULACIÓN DEL PROBLEMA..... | 14 |
| 1.2.3 JUSTIFICACIÓN | 14 |
| 1.3 OBJETIVOS..... | 15 |
| 1.3.1 General..... | 15 |
| 1.3.2 Específicos | 15 |
| 1.4 TIPO DE INVESTIGACIÓN | 16 |
| 1.5 ALCANCE..... | 16 |
| <i>CAPITULO II. MARCOS DE REFERENCIA</i> | <i>18</i> |
| 2.1 MARCO REFERENCIAL | 18 |
| 2.1.1 MARCO TEÓRICO | 18 |
| 2.1.2 Sistema de información para agendamiento, registro y control | 18 |
| 2.1.3 Metodología de desarrollo de software. | 19 |
| 2.1.4 Metodología de desarrollo SCRUM | 20 |
| 2.1.5 Procesos metodología de desarrollo SCRUM..... | 20 |
| 2.1.6 Roles | 20 |
| 2.1.7 Product Backlog / Blacklog del producto. | 21 |
| 2.1.8 Sprint Backlog. | 21 |
| 2.1.9 Sprint Planning Meeting / Planeación del Sprint. | 22 |
| 2.1.10 Daily Scrum o Stand-up Meeting | 22 |
| 2.1.11 Sprint Review / Revision del Sprint | 22 |

| | |
|---|------------------|
| 2.1.12 Sprint Retrospective / Retrospectiva del Sprint..... | 23 |
| 2.1.16 Aplicación Móvil..... | 23 |
| 2.1.17 JavaScript..... | 23 |
| 2.1.18 React Native..... | 24 |
| 2.1.19 Tienda de aplicaciones..... | 25 |
| 2.1.20 Bases de datos..... | 26 |
| 2.2 ESTADO DEL ARTE | 26 |
| 2.3 MARCO CONTEXTUAL..... | 28 |
| 2.3.1 Unad CCAV Pitalito | 28 |
| 2.3.2 Misión..... | 28 |
| 2.3.3 Visión..... | 29 |
| 2.4 MARCO LEGAL..... | 29 |
| 2.4.1 Ley estatutaria 1266 del 31 de diciembre de 2008..... | 29 |
| 2.4.2 Ley estatutaria 1581 de 2012 | 29 |
| 2.4.3 Ley estatutaria 1581 de 2012 | 29 |
| <i>CAPITULO III. INGENIERÍA DEL PROYECTO</i> | <i>30</i> |
| 3.1 METODOLOGÍA DE DESARROLLO | 30 |
| 3.2 PRODUCT BACKLOG | 30 |
| 3.2.1 Elementos imprescindibles..... | 30 |
| 3.2.2 Elementos importantes..... | 32 |
| 3.2.3 Diagramas de casos de uso | 33 |
| 3.2.4 Casos de uso en formato expandido..... | 34 |
| 3.3 ANÁLISIS Y DISEÑO PRELIMINAR..... | 36 |
| 3.4 DISEÑO DETALLADO | 37 |
| 3.5 IMPLEMENTACIÓN..... | 38 |
| 3.5.1 Modelado de datos | 38 |
| 3.5.2 Diagrama Entidad-Relación..... | 38 |

| | |
|--|------------------|
| 3.5.3 Arquitectura de Software | 39 |
| 3.5.4 Código Fuente | 41 |
| <i>CAPITULO IV. CONCLUSIONES Y RECOMENDACIONES.....</i> | <i>50</i> |
| 4.1 CONCLUSIONES | 50 |
| 4.2 RECOMENDACIONES | 52 |
| <i>CAPITULO V. BIBLIOGRAFIA.....</i> | <i>50</i> |

LISTA DE ILUSTRACIONES

| | |
|---|----|
| Ilustración 1. Tiendas de aplicaciones | 25 |
| Ilustración 2. Cuadro de resumen del estado del arte | 27 |
| Ilustración 3. Flujo de información del sistema | 32 |
| Ilustración 4. Casos de uso para el módulo de diccionario | 33 |
| Ilustración 5. Casos de uso modulo inscripcion a eventos | 34 |
| Ilustración 6. Casos de uso diccionario Unadista | 35 |
| Ilustración 7. Prototipo de base de datos | 36 |
| Ilustración 8. Maqueta aplicación | 37 |
| Ilustración 9. Diagrama de la arquitectura cliente-servidor..... | 39 |
| Ilustración 10. Diagrama de componentes del modelo-vista-controlador de Aplicación móvil | 40 |
| Ilustración 12. Vista login | 42 |
| Ilustración 13. Vista diccionario Unadista | 43 |
| Ilustración 14. Menu Aplicación | 44 |
| Ilustración 15. Geolocalización y eventos..... | 45 |
| Ilustración 16. Packages y plugins usados para el desarrollo | 46 |
| Ilustración 17. Archivo de rutas de la Aplicación | 47 |
| Ilustración 18. Conexión a laravel..... | 48 |
| Ilustración 19. Modelado de ingreso de texto en React Native | 49 |

GLOSARIO

App: Es el nombre usado comúnmente para referirse a las aplicaciones, que surge de acortar el vocablo inglés application. Es una pieza de software que se ejecuta en teléfonos móviles y tabletas y, como te habrás dado cuenta, es el objeto de estudio de este libro. Si aún no entiendes lo que es una app te recomendamos leer este libro con más atención.

Vista: Uno de los tres componentes del patrón modelo-vista-controlador. Se refieren a los archivos que contienen el código Java Script para la visualización de las vistas que se muestran al usuario.

Compilar: Es la acción de empaquetar un código. El resultado de compilar el código de una aplicación es el archivo final que está listo para ser subido a la tienda.

Controlador: Uno de los tres componentes del patrón modelo-vista-controlador. Representa el componente el cual define la lógica de la aplicación y cómo responde a las acciones del usuario. Actúa como intermediario entre la vista y el modelo.

Modelo: Uno de los tres componentes del patrón modelo-vista-controlador. Es el encargado de trabajar con los datos almacenados en la base de datos. Se encarga de realizar operaciones sobre la base de datos, tales como hacer consultas, insertar, modificar, entre otras, para si es necesario pasar los datos a los controladores.

Simulador: Un simulador permite probar la aplicación sin necesidad de contar con un móvil. De esta forma, se puede ejecutar el código en el ordenador y ver los resultados en la pantalla, con el fin de realizar comprobaciones preliminares sobre el funcionamiento de la app.

Ajax: Siglas de Asynchronous Javascript y XML

CSS: Siglas de cascading style sheets. Es el lenguaje utilizado para definir la presentación de documentos HTML.

Javascript: Es un lenguaje de programación para la elaboración de funciones sofisticadas en páginas web.

PHP: Lenguaje de programación de propósito general, usado principalmente para la creación de aplicaciones web del lado del servidor.

iOS (sistema operativo): iOS (anteriormente denominado iPhone OS) es un sistema operativo móvil de Apple desarrollado originalmente para el iPhone, siendo después usado en el iPod Touch e iPad. Es un derivado de Mac OS X, que a su vez está basado en Darwin BSD. iOS tiene 4 capas de abstracción: la capa del núcleo del sistema operativo, la capa de "Servicios Principales", la capa de "Medios de comunicación" y la capa de "Cocoa Touch". Todo el sistema se encuentra en la partición "/root" del dispositivo, ocupa poco menos de 500 megabytes.

Accesibilidad: Posibilidad de acceso a los contenidos por cualquier persona independientemente de sus capacidades físicas. A nivel visual está determinada, entre otras cosas, por el tamaño de los textos y botones y por el contraste que estos elementos tienen con el fondo. Una app accesible también hace referencia a una correcta programación del código que permite, por ejemplo, que los contenidos puedan ser interpretados por accesorios para ciegos.

Android: Es un sistema operativo diseñado para dispositivos de bolsillo. Android es un proyecto de código abierto y está soportado y desarrollado por Open Handset Alliance (OHA). Google y HTC son miembros de OHA, Google fue la primera compañía en empezar a trabajar con Android y HTC diseñó y fabricó la primera computadora de bolsillo con Android.

GPS: El sistema de posicionamiento global es un sistema de navegación por radio basado en satélite que consiste en tres segmentos - la constelación de satélites, la red de control desde tierra y el equipo del usuario. Las personas utilizan GPS en sus dispositivos móviles para aplicaciones de software de navegación.

INTRODUCCIÓN

En el presente documento se pretende explicar los motivos que nos inspiraron a llevar a cabo el desarrollo de la aplicación móvil y los objetivos que se pretenden alcanzar con el desarrollo de la misma.

Se presenta la definición del problema y su contexto, y la justificación del rol de las TIC en el apoyo de procesos para toma de decisiones y asistencia a eventos de los alumnos de la UNAD, para centralizar y automatizar ciertas tareas y para implementar el sistema de agendamiento, asistencia y conocimiento de la terminología, siglas y referencias Unadistas la cual es una estrategia para agilizar procesos y facilitar el acceso metadatos de la UNAD, todo esto como producto de las necesidades que se encontraron en CCAV Pitalito.

Se pone a consideración la aplicación Móvil AssistApp y las diferentes actividades que se llevan a cabo para su desarrollo, tales como la metodología de desarrollo, es decir SCRUM, y las diferentes etapas que conforman la metodología, es decir, el análisis de requerimientos, análisis y diseño preliminar, el diseño detallado y la implementación.

CAPITULO I. GENERALIDADES DEL PROYECTO

1.1 TÍTULO

Implementación del diccionario Unadista y desarrollo de la segunda fase del módulo AssistApp bajo una solución móvil que permita la georreferencia de usuarios y la inscripción a eventos del CCAV Pitalito

1.2 EL PROBLEMA

1.2.1 DESCRIPCIÓN DEL PROBLEMA.

Se ve la necesidad de buscar el desarrollo de herramientas prácticas que mejoren el proceso del registro de asistencias de todos los eventos presenciales a los que asisten estudiantes y docentes, participen durante el transcurso del año, esto además de ser una ventaja competitiva, permite una comunicación constante, teniendo siempre a la mano información y registros de los que se puede generar datos y estadísticas útiles para la universidad.

La UNAD usa un gran número de referencias y terminología que no son de uso general por lo que se hace difícil el aprendizaje y la comprensión de estos por parte de los estudiantes y demás personas cercanas a la institución. Sin una herramienta digital donde se puedan consultar cada uno de ellos el aprendizaje se hace lento... La necesidad de una herramienta ágil que cubra esta necesidad es evidente

Los estudiantes de la UNAD necesitan una forma que haga más simple el hecho de inscribirse a todo tipo de eventos que la universidad realiza un sistema al que se puedan ingresar de forma rápida y cómoda donde no solo se encuentra eventos de la universidad, sino que también un diccionario con toda la terminología y siglas propias de la universidad.

En las oficinas del CCAV Pitalito, el área de registro y control, se ofrecen tres tipos de servicios a los estudiantes, docentes u otra persona que esté vinculada a la institución. Estos tres servicios son de información y agenda de eventos, reserva e inscripción de eventos. Durante el proceso de atención a los usuarios se deben llenar una serie de formatos físicos, donde se registra información acerca de los mismos. De acuerdo con los datos consultados de los archivos de los informes mensuales, se encontró que el número de registro mes a mes es muy alto. Igualmente sucede con las clases y eventos llevados a cabo por los profesores para el separado de salas implementos necesarios para su clase o evento. Estos llevan a un proceso manual de programar con el personal de seguridad, el cual es difícil por el hecho de utilizar medios físicos y manuales para controlar la asistencia y orientación de los estudiantes.

Luego de haber realizado todos los procesos de registros en un determinado periodo, es necesario generar informes estadísticos que muestren el estado y el uso de todos los implementos y salas prestados en el CCAV Pitalito. Estos informes se hacen a partir de los registros físicos que se han almacenado durante todo un periodo donde se han brindado servicios a la comunidad universitaria. Dado que se manejan muchos registros físicos se hace difícil y lento el manejo de toda la información para así generar los informes estadísticos.

Este proceso al igual que ocurre en las oficinas del CCAV Pitalito, genera mucho gasto de papel y hace que el manejo de toda la información recolectada sea lento y desordenado.

1.2.2 FORMULACIÓN DEL PROBLEMA.

¿Cómo podría un sistema de información móvil agilizar los procesos de agendamiento y registro de asistencia a eventos académicos programados en el CCAV Pitalito y de qué manera ayudaría la implementación de un diccionario digital con terminología, siglas y referencias propias de la Unad a la comunidad educativa y demás beneficiarios de los servicios de la universidad en la región y su contexto?

1.2.3 JUSTIFICACIÓN

Sistematizar la información en cualquier organización que pretenda ser más eficiente en su servicio, requiere de una o varias herramientas tecnológicas que sean de apoyo para estructurar sus procesos y así hacerlos más competitivos. Por esta razón, la UNAD, desde el CCAV Pitalito, como ente de formación regional que se encuentra en constante crecimiento, presenta la necesidad de automatizar y optimizar su proceso de almacenamiento y seguimiento al registros de asistencias de los funcionarios y/o estudiantes que concurren a los diferentes eventos presenciales programados en las instalaciones de la institución, en cada periodo académico, a través de una solución informática, ambiente web y móvil, que permita regular y controlar de forma sencilla, eficaz y confiable la información mencionada.

En este proyecto se construirá una aplicación móvil para Android y otra para IOS para dar agilidad a los procesos de revisión e inscripción a eventos de los estudiantes de la UNAD y tener certeza absoluta sobre horarios y disponibilidad de las diferentes actividades que la universidad realiza, así como también un diccionario con toda la terminología, siglas y demás temas relacionados con la universidad.

1.3 OBJETIVOS

1.3.1 General

Desarrollar una aplicación móvil, para Android y iOS, que permita el registro a los eventos académicos programados por el centro educativo más cercano a la ubicación de los usuarios de la aplicación y poder consultar y apropiarse sobre la terminología o conceptos propios de la UNAD y de la educación a distancia.

1.3.2 Específicos

- Identificar las falencias que actualmente se presentan en el proceso de almacenamiento de los registros de asistencia de los diferentes eventos desarrollados en las instalaciones del CCAV Pitalito por observación directa.
- Diseñar una aplicación móvil bajo la programación React Native de tal manera que permita coleccionar los registros de asistencia, validarlos y generar los respectivos informes y estadísticas.

- Implementar una interfaz móvil, de la aplicación web AssistApp, para las plataformas Android y iOS que permita identificar qué actividades o eventos académicos existen programados en el centro educativo más cercano a la ubicación del usuario de la aplicación.
- Incluir el módulo del “Diccionario Unadista” con la definición de los términos propios de la institución sobre la interfaz móvil.

1.4 TIPO DE INVESTIGACIÓN

Cualitativa, descriptiva

Desde la dirección del CCAV Pitalito se establecen estrategias de seguimiento y acompañamiento para los estudiantes matriculados, estas actividades, de tipo académico, son programadas y orientadas desde la consejería estudiantil y apoyadas por los tutores como estrategia de retención y permanencia de los estudiantes en la universidad, con los cuales se les brinda un servicio de atención a estudiantes, docentes y funcionarios. De esta forma se generan documentos con contenidos que llevan un registro de todas las clases eventos y reuniones programados en el centro.

Así mismo, en el CCAV Pitalito los estudiantes deben verificar varios canales de información asignados por la universidad donde reciben información enviada por los tutores y administrativos para establecer si el estudiante puede asistir a los eventos agendados. Se busca especificar las propiedades de todos aquellos procedimientos que son realizados en las oficinas del CCAV Pitalito mediante un análisis de estos. Para ello se selecciona una serie de actividades y se mide cada una de ellas independientemente, Con el fin de poder adoptar estrategias para mitigar la deserción estudiantil.

1.5 ALCANCE

Para el desarrollo del proyecto se implementará un conjunto adecuado de controles, que abarcan políticas, prácticas, procedimientos, estructuras organizacionales y funciones de software como lo son:

- Modulo diccionario Unadista.
 - Buscador avanzado
 - Vista multimedia
- Modulo Geolocalización y agenda
- Modulo interfaz de usuarios
 - Iniciar sesión
 - Registro de usuario

CAPITULO II. MARCOS DE REFERENCIA

2.1 MARCO REFERENCIAL

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa (ROI). Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación¹.

2.1.1 MARCO TEÓRICO

Se presenta a continuación el marco teórico relacionado con los sistemas de información para agendamiento, registro, control y metodología de desarrollo de software SCRUM.

2.1.2 Sistema de información para agendamiento, registro y control

En un sentido general, un sistema de información (SI) es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su uso posterior, generados para cubrir una necesidad o un objetivo². El sistema de información de la Agenda de eventos y diccionario Unadista busca fortalecer procesos interactivos entre en centro y los estudiantes mejorando la participación académica y fortaleciendo el conocimiento de los

¹ Metodología Scrum. (2008). *Softeng*. on <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>

² O'Leary, Timothy y Linda. (2008). *Computing Essentials Introductory 2008*. McGraw-Hill on Computing2008.com

metadatos de la institución, promoviendo en los estudiantes procesos participativos de búsqueda, registro, análisis y uso de la información para la gestión de asistencia a eventos.

Los sistemas de información para agenda tienen como componente principal los registros de eventos electrónicos, los cuales son procesados por un sistema de información apoyado por una base de datos para asegurar su disponibilidad. Un registro de evento es una versión digital de los registros de eventos de un profesor administrativo o estudiante. Los registros de eventos son procesados en tiempo real lo cual hace que la información esté disponible de manera casi instantánea y segura a usuarios interesados.

Un sistema de agendamiento está diseñado para ir más allá de simples registros, ya que puede dar a los estudiantes una participación más amplia en los diferentes procesos de la universidad. Los registros pueden mostrar eventos, laboratorios, clases, congresos, entre otros. Una de las ventajas es el no uso de papel y el orden en información y reportes, aumentan la productividad, apoyar la toma de decisiones, permitir el acceso inmediato a la información.

2.1.3 Metodología de desarrollo de software.

Una Metodología de desarrollo de software, consiste principalmente en hacer uso de diversas herramientas, técnicas, métodos y modelos para el desarrollo. Regularmente este tipo de metodología, tienen la necesidad de venir documentadas, para que los programadores que estarán dentro de la planeación del proyecto, comprendan perfectamente la metodología y en algunos casos el ciclo de vida del software que se pretende seguir.

Aunque actualmente existen mucha variedad en metodologías de programación. La realidad es que todas están basadas en ciertos enfoques generalistas que se crearon hace muchos años, algunos tipos de metodologías de desarrollo de software que se utilizaron e inventaron al principio de nuestra era tecnológica y son las que veremos a continuación³.

³ OK HOSTING. Metodologías de desarrollo de software. [Citado Septiembre 10 de 2018]. Disponible en internet: < https://okhosting.com/blog/metodologias-del-desarrollo-de-software/#En_que_consisten_las_Metodologias_de_Desarrollo_de_Software>.

2.1.4 Metodología de desarrollo SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto⁴.

2.1.5 Procesos metodología de desarrollo SCRUM

La metodología Scrum, es bastante amigable y fomenta lo que es el trabajo en equipo en todo momento, con la finalidad de conseguir los objetivos de una forma rápida. Veamos ahora cuales son los procesos con los cuales funciona la metodología⁵.

2.1.6 Roles

El equipo de Scrum consiste en tres diferentes roles:

⁴ PROYECTOS AGILES. Que es SCRUM [En línea]. [[Citado Septiembre 10 de 2018]]. Disponible en internet: <<https://proyectosagiles.org/que-es-scrum/>>

⁵ OK HOSTING. Metodologías de desarrollo de software. [Citado Septiembre 10 de 2018]. Disponible en internet: <https://okhosting.com/blog/metodologias-del-desarrollo-de-software/#En_que_consisten_las_Metodologias_de_Desarrollo_de_Software>.

- El *Product Owner/Dueño* del producto. es la “voz del cliente” y el responsable de desarrollar, mantener y priorizar las tareas en el *backlog*.
- El *Scrum Master* es responsable de asegurarse que el trabajo del equipo vaya bien siguiendo las bases de Scrum. Además, se encarga de remover cualquier obstáculo que pueda encontrar el equipo de desarrollo.
- Los *Development Team Members/Miembros del Equipo* de desarrollo son los encargados de escribir y probar el código.

2.1.7 Product Backlog / Blacklog del producto.

El Product Backlog no es más que una lista de las funcionalidades del producto a desarrollar. Sin embargo, no se trata de una lista cualquiera hecha con escritos y nada más. El Product Backlog debe estar ordenado de acuerdo a las prioridades del sistema de más a menos, con la idea de que las cosas con mayor prioridad sean las que se realicen antes de cualquier cosa. De forma concreta, digamos que el objetivo base del Product Owner, es que nos dé respuesta a la pregunta “¿Qué hay que hacer?”⁶.

Esto puede referirse a todo elemento que sea parte del proyecto: puede ser un bug, una referencia o parte de un requerimiento. Brindan información muy general del proyecto y muchas veces no son tomados como requerimientos oficiales⁷.

2.1.8 Sprint Backlog.

Una vez que ya contamos con el Product Backlog terminado, entonces aparecerá el primer Sprint Backlog. Pero ¿Qué es el Sprint Backlog? Consiste básicamente en seleccionar algunos de los puntos escritos en el Product Backlog, los cuales procederán a ser realizados. Sin embargo, en este punto el Sprint Backlog tiene como requisito marcar el tiempo en que se llevará a cabo el Sprint⁸.

⁶ OK HOSTING. Metodologías de desarrollo de software. [Citado Septiembre 10 de 2018]. Disponible en internet: < https://okhosting.com/blog/metodologias-del-desarrollo-de-software/#En_que_consisten_las_Metodologias_de_Desarrollo_de_Software>.

⁷ OK HOSTING. Metodologías de desarrollo de software. [Citado Septiembre 10 de 2018]. Disponible en internet: < https://okhosting.com/blog/metodologias-del-desarrollo-de-software/#En_que_consisten_las_Metodologias_de_Desarrollo_de_Software>.

⁸ PLATZI. Como funciona la metodología SCRUM. [Citado Septiembre 10 de 2018]. Disponible en internet: <<https://platzi.com/blog/metodologia-scrum-fases/>>.

Todos los involucrados en el equipo se reúnen para planificar el Sprint. Durante este evento se decide qué requerimientos o tareas se le asignará a cada uno de los elementos del equipo. Cada integrante deberá asignar el tiempo que crea prudente para llevar a cabo sus requerimientos. De esta manera se define el tiempo de duración del Sprint.⁹

2.1.9 Sprint Planning Meeting / Planeación del Sprint.

Antes de iniciar un Sprint, el cual es la fase de desarrollo, se realiza lo que es un Sprint Planning Meeting. En este proceso del Scrum, es una reunión que se realiza para definir plazos y procesos a efectuarse para el proyecto establecido en el Product Backlog. Algo importante que debes saber, es que cada Sprint, se compone de diversos features, que no son otra cosa más que procesos o subprocesos que se deben realizar, puede ser la creación de un logo, la gestión de contenido, el diseño visual, etc. Todo dependerá del proceso que se desee llevar a cabo.

2.1.10 Daily Scrum o Stand-up Meeting

Cuando un Sprint está en proceso, después de haber hecho la planeación del proyecto mediante plazos y procesos, entonces entramos a lo que son los Daily Scrum o Stand-up Meeting. Aquí básicamente lo que se hace son reuniones diarias mientras se está llevando a cabo un Sprint, para responder las siguientes preguntas: ¿Que hice ayer?, ¿Qué voy a hacer hoy, ¿Qué ayuda necesito? Aquí entra en función el Scrum Master, un puesto que igual más adelante les explicaré. Pero el será el encargado de determinar la solución de los problemas y cada complicación que suceda.

2.1.11 Sprint Review / Revision del Sprint

El Sprint Review, es básicamente una reseña de lo que fue el Sprint. Consiste específicamente en la revisión del Sprint terminado y para este punto ya tendría

⁹ PLATZI. Como funciona la metodología SRUM. [Citado Septiembre 10 de 2018]. Disponible en internet: <<https://platzi.com/blog/metodologia-scrum-fases/>>.

que haber algo que mostrarle al cliente, algo realmente visual o tangible para que se pueda analizar un cierto avance.¹⁰

Los miembros del equipo y los clientes se reúnen para mostrar el trabajo de desarrollo de software que se ha completado. Se hace una demostración de todos los requerimientos finalizados dentro del Sprint.¹¹

2.1.12 Sprint Retrospective / Retrospectiva del Sprint

Para concluir, el Sprint Retrospective, permite al equipo analizar los objetivos cumplidos, si se cometieron errores, visualizarlos y tratar de no cometerlos nuevamente más adelante. Básicamente también sirve este proceso para lo que son la implementación de mejoras.

En este evento el Product Owner se reúne con todo su equipo de trabajo y su Scrum Master para hablar sobre lo ocurrido durante el Sprint. Los puntos principales a tratar en esta reunión son:¹².

- Qué se hizo mal durante el Sprint para poder mejorar el próximo.
- Qué se hizo bien para seguir en la misma senda del éxito.
- Qué inconvenientes se encontraron y no permitieron poder avanzar como se tenía planificado.

2.1.16 Aplicación Móvil.

Una aplicación Móvil es una aplicación diseñada para dispositivos móviles que está alojada en tiendas de aplicaciones como App Store y Play store de manera remota accesible a todos los usuarios que deseen descargarla en tiempo real. Actualizable y dinámica gracias al uso de internet.

2.1.17 JavaScript.

¹⁰ PLATZI. Como funciona la metodología SCRUM. [Citado Septiembre 10 de 2018]. Disponible en internet: <<https://platzi.com/blog/metodologia-scrum-fases/>>.

¹¹ OK HOSTING. Metodologías de desarrollo de software. [Citado Septiembre 10 de 2018]. Disponible en internet: <https://okhosting.com/blog/metodologias-del-desarrollo-de-software/#En_que_consisten_las_Metodologias_de_Desarrollo_de_Software>.

¹² PLATZI. Como funciona la metodología SCRUM. [Citado Septiembre 10 de 2018]. Disponible en internet: <<https://platzi.com/blog/metodologia-scrum-fases/>>.

JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMA Script. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico¹³. ... Todos los navegadores y dispositivos modernos interpretan el código JavaScript.

2.1.18 React Native.

React-Native es una plataforma desarrollada por Facebook, open source para el desarrollo de aplicaciones móviles cross-platform completamente nativas utilizando React y por lo tanto nuestro amado JavaScript ;) Al momento de escribir este artículo podemos crear aplicaciones para IOS, Android y Ubuntu.

Para empezar a crear aplicaciones móviles con React-Native tenemos que tener en mente¹⁴:

1. React-Native no es un Webview, todo lo que se ejecuta es absolutamente nativo ;)
2. Nuestras aplicaciones no corren en navegador por lo tanto no tenemos DOM, CSS, etc.
3. Al no tener CSS ¿Cómo hacemos el layout de nuestra aplicación? React-Native soluciona este problema utilizando un subconjunto de propiedades de CSS.
4. La aplicación al estar escrita en JavaScript, nos permite tener acceso a la gran cantidad de módulos que hay a nuestra disposición en los gestores de paquete con NPM. La única restricción es que las librerías no dependan del Navegador o de Node.js.
5. Tenemos que usar React y la manera de hacer las cosas a la React, lo cual es bueno.
6. React-Native nos ofrece algunos polyfills o shims para Fetch, Sockets etc.
7. Probar una app hecha en React-Native es como probar cualquier código JS.

¹³ WIKIPEDIA. JavaScript [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: <<https://es.wikipedia.org/wiki/JavaScript>>

¹⁴ MEDIUM, Ale Garcia. React Native [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: <<https://medium.com/monoku/react-native-est%C3%A1-cambiando-la-forma-en-la-que-desarrollamos-aplicaciones-m%C3%B3viles-de7bbb742555>>

2.1.19 Tienda de aplicaciones

Ilustración 1. Tiendas de aplicaciones



Imagen tomada de: <https://blog.goldenmac.info/como-pasar-aplicaciones-de-android-a-iphone/>

La tienda es el lugar donde se aloje la aplicación para que esté disponible para descargar por todos los usuarios interesados. Existen diferentes tipos de tiendas para descargar aplicaciones, estas pueden ser creadas por el mismo sistema operativo o por independientes. Las tiendas organizan las aplicaciones y cada una tiene normas diferentes de retribución y publicación. Para la distribución de aplicaciones móviles existen diferentes plataformas distribuidoras¹⁵:

Google Play: Google Play (anteriormente Android Market) es una plataforma de distribución de software en línea desarrollado por [Google Inc.](#) para dispositivos con sistema operativo [Android](#). Fue lanzado en octubre de 2008. Hasta octubre de 2012, Google Play contaba con más de 700 000 aplicaciones ¹⁴ En la plataforma se encuentran disponibles tanto aplicaciones gratuitas como de pago¹⁶.

App Store: La App Store fue el primer servicio de distribución de aplicaciones, siendo lanzada el 10 de julio de 2008. En 2016, el CEO de [Apple](#), [Tim Cook](#), anunció que existen 2.000.000 aplicaciones disponibles para dispositivos con [iOS](#).¹⁵ Desde su creación en 2008, más de un millón de aplicaciones estuvieron disponibles en el App Store. Numerosas empresas utilizan este canal para distribuir las aplicaciones colaborativas, de gestión y de productividad a los usuarios externos e internos.

¹⁵ WIKIPEDIA, Aplicacion Movil [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: <https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_m%C3%B3vil#Tiendas_de_aplicaciones>

¹⁶ WIKIPEDIA, Aplicacion Movil [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: <https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_m%C3%B3vil#Tiendas_de_aplicaciones>

En julio de 2012, Apple creó *App Store Volume purchasing for business*. Disponible únicamente en EE. UU., este programa permite a las empresas comprar aplicaciones en grandes cantidades con el fin de distribuirlas a sus colaboradores a través de códigos promocionales. Es posible también integrar en esta tienda "business to business", aplicaciones desarrolladas por terceros y que no son publicadas en el App Store clásico¹⁷.

2.1.20 Bases de datos.

Una base de datos es una colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos es un sistema de archivos electrónico.

Las bases de datos tradicionales se organizan por campos, registros y archivos. Un campo es una pieza única de información; un registro es un sistema completo de campos; y un archivo es una colección de registros. Por ejemplo, una guía de teléfono es análoga a un archivo. Contiene una lista de registros, cada uno de los cuales consiste en tres campos: nombre, dirección, y número de teléfono¹⁸.

2.2 ESTADO DEL ARTE

El contenido de la literatura revisada se encuentra en el Anexo F. Estado del arte.

¹⁷ WIKIPEDIA, Aplicacion Movil [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: <https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_m%C3%B3vil#Tiendas_de_aplicaciones>

¹⁸ MASADELANTE. ¿Que es una base de datos? [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: < <http://www.masadelante.com/faqs/base-de-datos>>

Tabla 1. Cuadro de resumen del estado del arte

| Título | Metodología de desarrollo | Framework | Tecnología |
|--|----------------------------|-----------|-----------------------|
| UniMóvil aplicación para dispositivos móviles -celulares y/o tabletas- que permita el acceso de alumnos y docentes a la información de la universidad. | No especifica | Ninguno | PHP, JavaScript, Java |
| Aplicación móvil como recurso de apoyo en los estudiantes de la licenciatura en Tic aplicadas a la educación modalidad a distancia | Modelo “Lineal secuencial” | Ninguno | Android Studio |
| Aplicación móvil para el acceso a la información de la Universidad de Cantabria | SCRUM | Ninguno | Android Studio |
| Squid | No especifica | Ninguno | Andorid Studio |
| Noteshelf | Iconix | Ninguna | Xcode |

Fuente: Universia Colombia (2017) - mejores aplicaciones para estudiantes
<https://noticias.universia.net.co/cultura/noticia/2016/01/29/1135891/25-mejores-aplicaciones-estudiantes.html>

En el siguiente punto resumiré la estructura revisada en lo referente a proyectos de desarrollo de software cuyo objetivo ha sido el desarrollo de un sistema una aplicación móvil para mejorar el bienestar institucional, al bienestar universitario y todos los actores que forman parte del CCAV Pitalito. También se revisaron proyectos donde se haya hecho uso de la metodología de desarrollo Scrum y el Framework React Native.

Como se puede ver en el cuadro anterior, existen proyectos previamente elaborados donde se ha desarrollado una aplicación para móvil para apoyar las labores de una oficina de instituciones universitarias, y proyectos donde se ha usado la metodología de desarrollo SCRUM o donde se ha usado el Framework React Native.

2.3 MARCO CONTEXTUAL.

2.3.1 Unad CCAV Pitalito

La Universidad Nacional Abierta y a Distancia es una universidad privada, estatal de carácter nacional, financiada por el Estado Colombiano y por recursos propios. La sede principal se encuentra la ciudad de Bogotá D.C, se denomina José Celestino Mutis. Cuenta actualmente con 75.000 estudiantes, de los cuales 60.000 cursan programas de educación superior y 15.000 desarrollan los ciclos previos de alfabetización, básica primaria y bachillerato a distancia¹⁹.

2.3.2 Misión

La Universidad Nacional Abierta y a Distancia (UNAD) tiene como misión contribuir a la educación para todos a través de la modalidad abierta, a distancia y en ambientes virtuales de aprendizaje, mediante la acción pedagógica, la proyección social, el desarrollo regional y la proyección comunitaria, la inclusión, la investigación, la internacionalización y las innovaciones metodológicas y didácticas, con la utilización de las tecnologías de la información y las comunicaciones para fomentar y acompañar el aprendizaje autónomo, generador de cultura y espíritu emprendedor que, en el marco de la sociedad global y del conocimiento, propicie el desarrollo económico, social y humano sostenible de las comunidades locales, regionales y globales con calidad, eficiencia y equidad social²⁰.

¹⁹ Unad. Universidad nacional abierta y a distancia [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: < <https://www.unad.edu.co/>>

²⁰ Unad. Universidad nacional abierta y a distancia [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: < <https://www.unad.edu.co/>>

2.3.3 Visión.

Se proyecta como una organización líder en Educación Abierta y a Distancia, reconocida a nivel nacional e internacional por la calidad innovadora y pertinencia de sus ofertas y servicios educativos y por su compromiso y aporte de su comunidad académica al desarrollo humano sostenible, de las comunidades locales y globales²¹.

2.4 MARCO LEGAL

De acuerdo a las actividades realizadas por el área de registro y control, se debe cumplir con la siguiente normatividad:

2.4.1 Ley estatutaria 1266 del 31 de diciembre de 2008

Por la cual se dictan las disposiciones generales del Hábeas Data y se regula el manejo de la información contenida en bases de datos personales, en especial la financiera, crediticia, comercial, de servicios y la proveniente de terceros países y se dictan otras disposiciones²².

2.4.2 Ley estatutaria 1581 de 2012

Entró en vigencia la Ley 1581 del 17 de octubre 2012 de PROTECCIÓN DE DATOS PERSONALES, sancionada siguiendo los lineamientos establecidos por el Congreso de la República y la Sentencia C-748 de 2011 de la Corte Constitucional.

2.4.3 Ley estatutaria 1581 de 2012

Por la cual se dictan disposiciones generales para la protección de datos personales.

²¹ Unad. Universidad nacional abierta y a distancia [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: < <https://www.unad.edu.co/> >

²² Seguridad de la informacion [En línea]. [Citado 10 de septiembre de 2018]. Disponible en internet: < <http://seguridadinformacioncolombia.blogspot.com/2010/02/marco-legal-de-seguridad-de-la.html> >

CAPITULO III. INGENIERÍA DEL PROYECTO

3.1 METODOLOGÍA DE DESARROLLO

Para el desarrollo de la aplicación Móvil, se decide usar la metodología de desarrollo de software SCRUM. A continuación, se describe el desarrollo de cada uno de los procesos de SCRUM durante el desarrollo de la aplicación.

3.2 PRODUCT BACKLOG

A continuación, se describen la forma en que se llevó a cabo para hacer lista de las funcionalidades del producto a desarrollar.

3.2.1 Elementos imprescindibles

En este apartado se apuntan las tareas que son consustanciales al proyecto, es decir, que no pueden obviarse porque pondrían en riesgo la ejecución de este. Por lo general, se expresan como objetivos a gran escala. Para esto se realizan entrevistas a personal administrativo, profesores y estudiantes del CCAV Pitalito. En la siguiente ilustración se muestra un ejemplo de un requerimiento funcional que se definió para el desarrollo de la aplicación:

Tabla 3. Requerimiento número 1.

| | |
|--------------------------------------|--|
| Requerimiento No. | 1 |
| Módulo | Diccionario Unadista |
| Usuario | Estudiante |
| Descripción del requerimiento | <p>El sistema debe permitir que por medio de la aplicación el estudiante pueda buscar terminología, siglas y referencias usadas por la institución que por lo general son completamente desconocidos por los estudiantes. Hacer un buscador avanzado que atreves de la base de datos traiga descripción y vínculos sobre el significado de las palabras buscadas especificando los siguientes datos:</p> <ul style="list-style-type: none"> • Palabra, sigla o referencia <p>Durante la elección significado, el sistema mostrará la descripción de este.</p> |

Fuente: autor: Sebastián Gamboa

Tabla 4. Requerimiento número 2.

| | |
|--------------------------------------|---|
| Requerimiento No. | 2 |
| Módulo | Inscripción a eventos |
| Usuario | Estudiante |
| Descripción del requerimiento | <p>El sistema debe mostrar agenda de eventos para el programa en que se encuentra matriculado el estudiante, así como una interfaz de geolocalización para mostrar los eventos del centro más cercano y un medio para la inscribirse y asistir y cualquiera de estos.</p> |

Fuente: autor: Sebastián Gamboa

3.2.2 Elementos importantes

Después de hacer una lista de los requerimientos, se procede a elaborar el modelo de flujo de información, que puede ser modificado dependiendo de cómo avanza el proyecto.

Fuente: Proyecto PIE “Sistema de Información para la Administración y Gestión de Eventos y Escenarios Académicos en el CCAV de Pitalito - AssistApp” – Jhon Perez

En la ilustración anterior el modelo de flujo de información se elabora de acuerdo con lo especificado en los requerimientos y teniendo en cuenta los avances de la primera fase.

Ilustración 5. Nube de Servicios de la plataforma web AssistApp

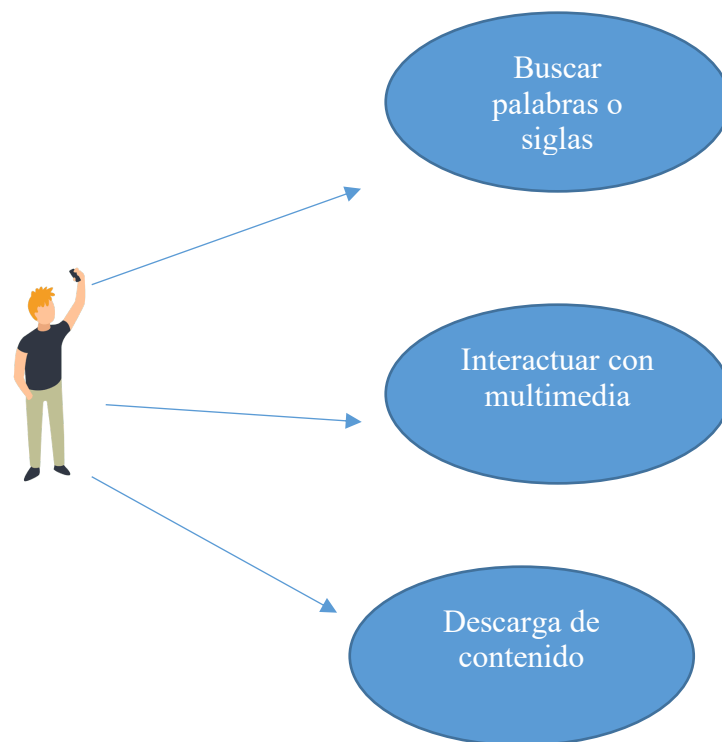


Fuente: Proyecto PIE “Sistema de Información para la Administración y Gestión de Eventos y Escenarios Académicos en el CCAV de Pitalito - AssistApp” – Jhon Perez

3.2.3 Diagramas de casos de uso

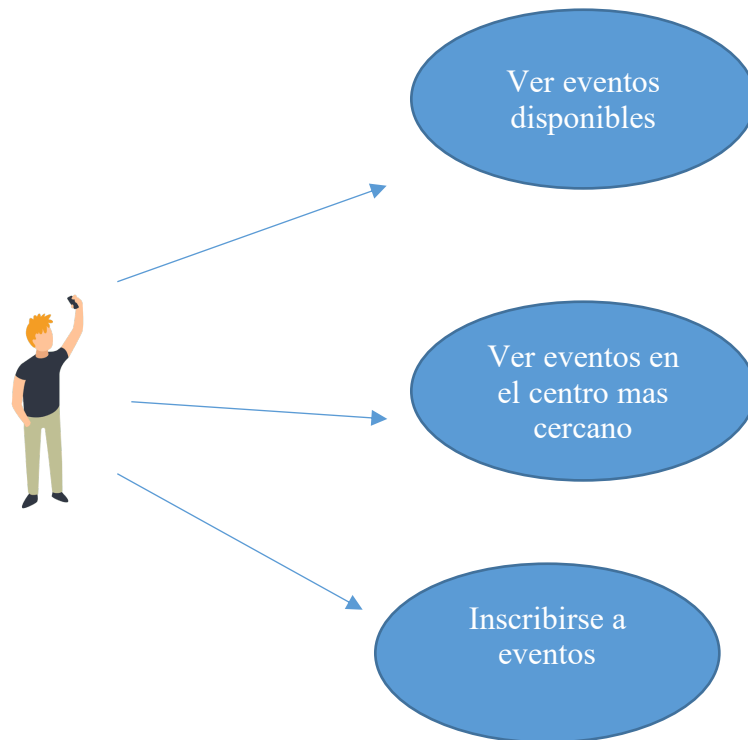
Ahora se van a describir los casos de uso de la aplicación. En esta aplicación el único usuario involucrado será el estudiante ya que en la primera fase que es la web cubre la parte administrativa donde se crean y gestionan los eventos. La siguiente ilustración muestra el diagrama de casos de uso para la aplicación móvil AssistApp:

Ilustración 6. Casos de uso para el módulo de diccionario



Fuente: autor: Sebastián Gamboa

Ilustración 7. Casos de uso modulo inscripción a eventos



Fuente: autor: Sebastián Gamboa

La primera de las dos ilustraciones anteriores muestra los casos de uso de manera general del módulo de del diccionario Unadista. El segundo diagrama muestra casos de uso para el módulo de inscripción a eventos de la universidad.

3.2.4 Casos de uso en formato expandido

En esta etapa se elaboran los casos de uso en formato expandido, los cuales describen de manera más detallada las actividades que conforman un caso de uso en particular, es decir, paso a paso, junto con sus vías alternativas. A continuación, se muestra el caso de uso en formato para el módulo de diccionario Unadista:

Tabla 4. Casos de uso diccionario Unadista

| | |
|--|--|
| Caso de Uso | Buscar términos o siglas |
| Actores | Estudiante |
| Propósito | Conocer el significado de términos que solo son usados en la UNAD |
| Tipo | Principal |
| Descripción | Un estudiante matriculado en la UNAD puede ingresar al sistema con su número de cedula y mediante un buscador puede encontrar términos institucionales |
| Referencias cruzadas | |
| Curso normal de los eventos | |
| Acción de los actores | Respuesta del sistema |
| 1. El estudiante escribe la palabra que quiere buscar | La aplicación despliega una lista con las palabras encontradas en la base de datos que coinciden con lo ingresado en el buscador |
| 2. El estudiante selecciona el resultado que se ajusta a su búsqueda | La aplicación muestra una descripción, contenido multimedia y un enlace donde puede encontrar más información |
| Cursos alternativos | |
| Acción 2: Si el estudiante ingresa datos inválidos la aplicación mostrara una alerta avisando de la inconsistencia en la información | |

Fuente: Los autores.

En la ilustración anterior, como se puede ver el caso de uso en formato expandido para el caso de uso buscar en Diccionario Unadista es descrito más

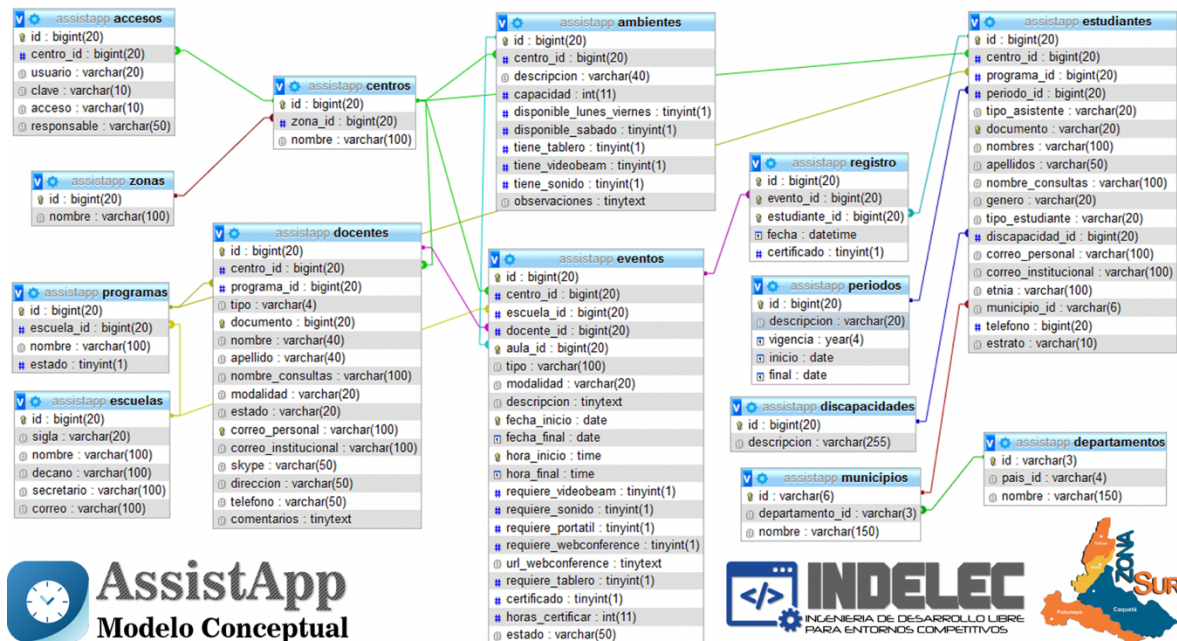
detalladamente, mostrando cada paso de manera secuencial y también las vías alternativas.

3.3 ANÁLISIS Y DISEÑO PRELIMINAR

En esta etapa del proceso, se realiza un prototipo de datos para cada uno de los casos de uso definidos previamente. En el prototipo tenemos una visualización de los pasos que ocurren para todos los casos de uso.

Así mismo, se actualiza el prototipo de datos de la aplicación. Haciendo esto se pueden descubrir clases nuevas, corregir ambigüedades y se añaden atributos a los objetos mencionados.

Ilustración 10. Prototipo de base de datos



Fuente: Sebastián Gamboa. María DB.

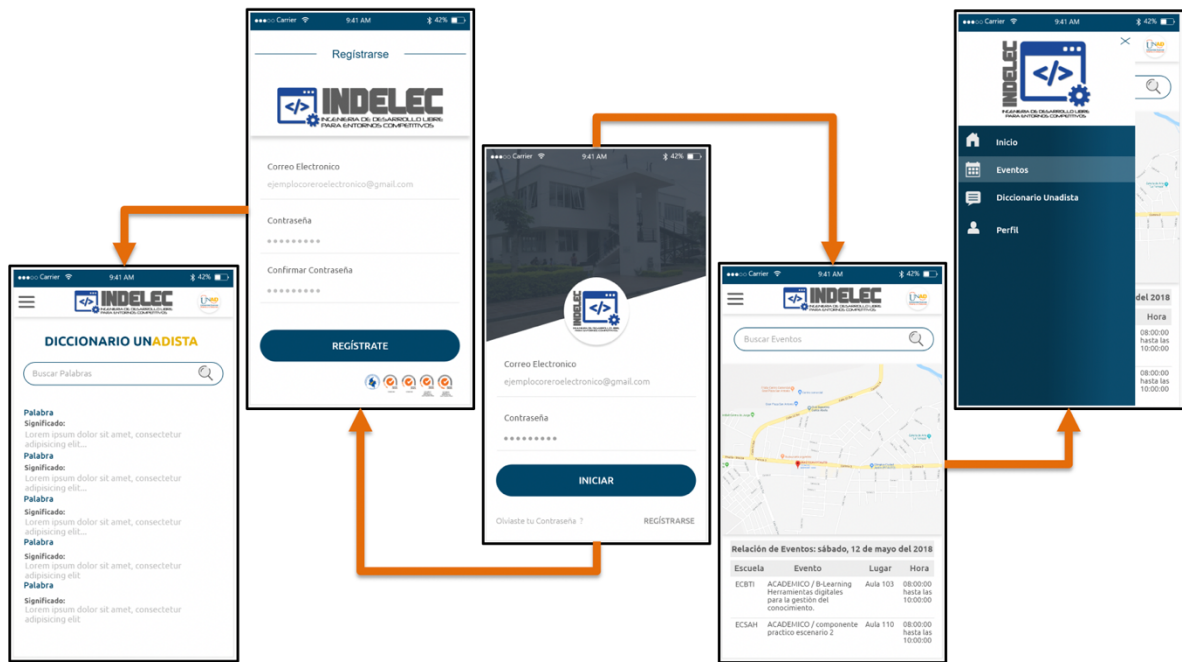
En este caso, el prototipo de datos muestra los diferentes elementos involucrados en los casos de uso. La secuencia muestra que primero a través de la web hecha en la primera fase se crearan lo usuarios administradores que son

los encargados de crear y administrar eventos en la UNAD y los estudiantes que tienen acceso a la aplicación. Luego a través de la aplicación móvil se consulta los datos y se muestran al estudiante.

3.4 DISEÑO DETALLADO

En esta etapa del proceso, se procede a realizar una maqueta de lo que será la aplicación y la secuencia para cada caso de uso descrito anteriormente el modelo se usará por Android y para IOS.

Ilustración 11. Maqueta aplicación



Fuente: Sebastián Gamboa. Software Adobe XD.

La maqueta muestra las vistas y módulos de operaciones de la aplicación primero lo usuarios se encontrarán con una interfaz de usuarios donde pueden ingresar y registrarse. Cuando inicien sesión lo primero que verán será el diccionario Unadista y a través de un menú desplegable encontraran la opción de ver los eventos interactuar y escribirse en los mismos como también la

georreferencia donde va a mostrar los centros más cercanos al usuario y como prioridad traerá los datos de eventos a realizar en este centro.

3.5 IMPLEMENTACIÓN

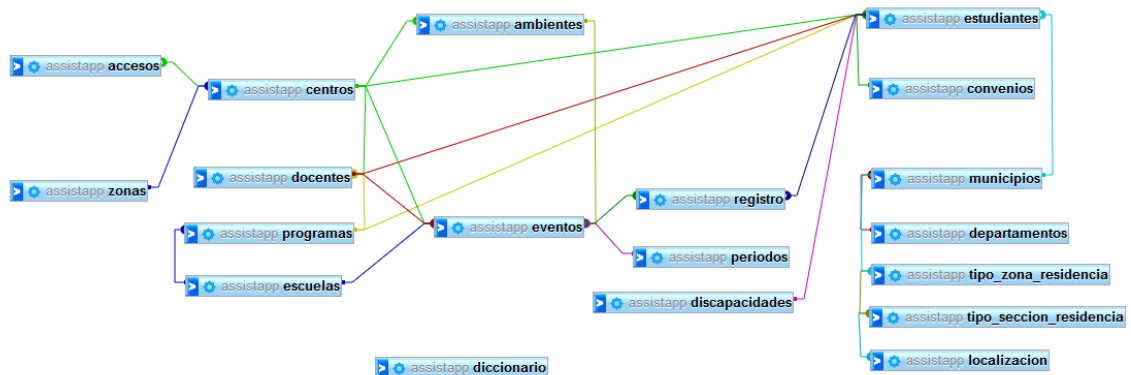
3.5.1 Modelado de datos

En el modelo de datos se obtiene la información necesaria para crear un flujo de secuencias para el almacenado y administración de la información en esta aplicación los datos deben sincronizarse con la fase 1 de este proyecto se agregan las nuevas tablas y se crean relaciones para hacer un flujo de información dinámico y rápido.

3.5.2 Diagrama Entidad-Relación

El diagrama entidad-relación se obtiene a partir del diagrama de datos obtenido durante las agregaciones y modificaciones que se hacen al modelo de datos el cual se convierte en un diagrama de clases conforme se avanza en el proyecto. Los diagramas de clases contienen los atributos necesarios para generar el modelo de base de datos. El diagrama entidad-relación se elaboró usando el software Maria DB.

Ilustración 12. Modelo entidad relación

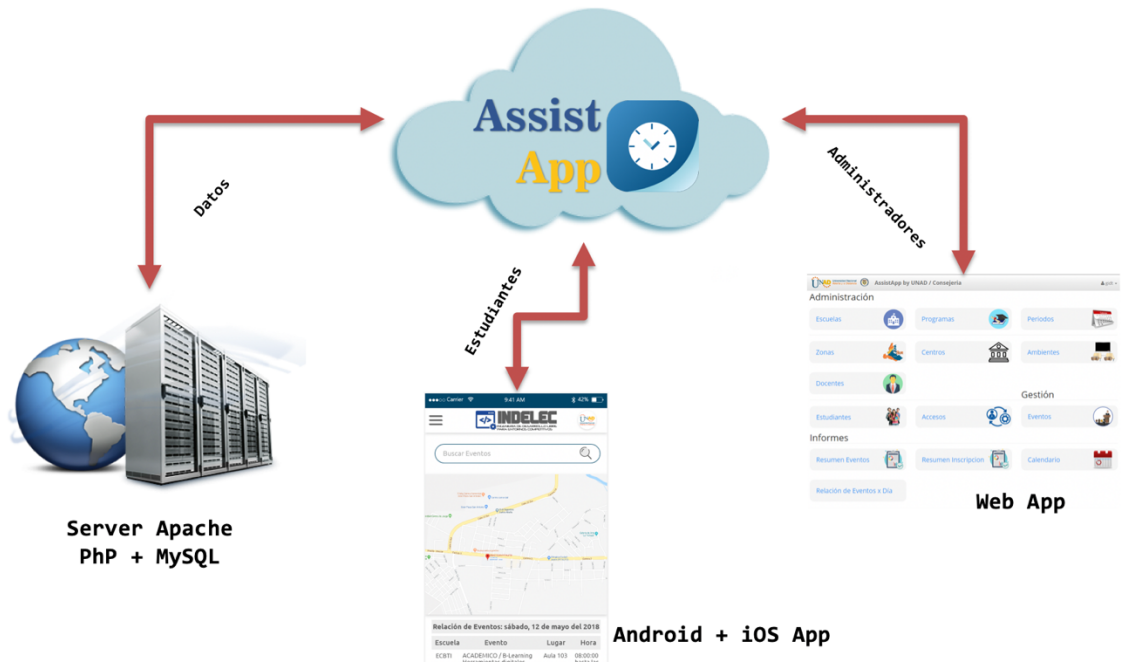


Fuente: Sebastián Gamboa. María DB

3.5.3 Arquitectura de Software

Con los previos pasos anteriores realizados, el siguiente paso es la definición de la arquitectura del de servidores y plataformas de almacenamiento. En este caso, la arquitectura cliente servidor es usada para la implementación de la aplicación móvil en la red, y se ilustra en el siguiente diagrama:

Ilustración 13. Diagrama de la arquitectura cliente-servidor



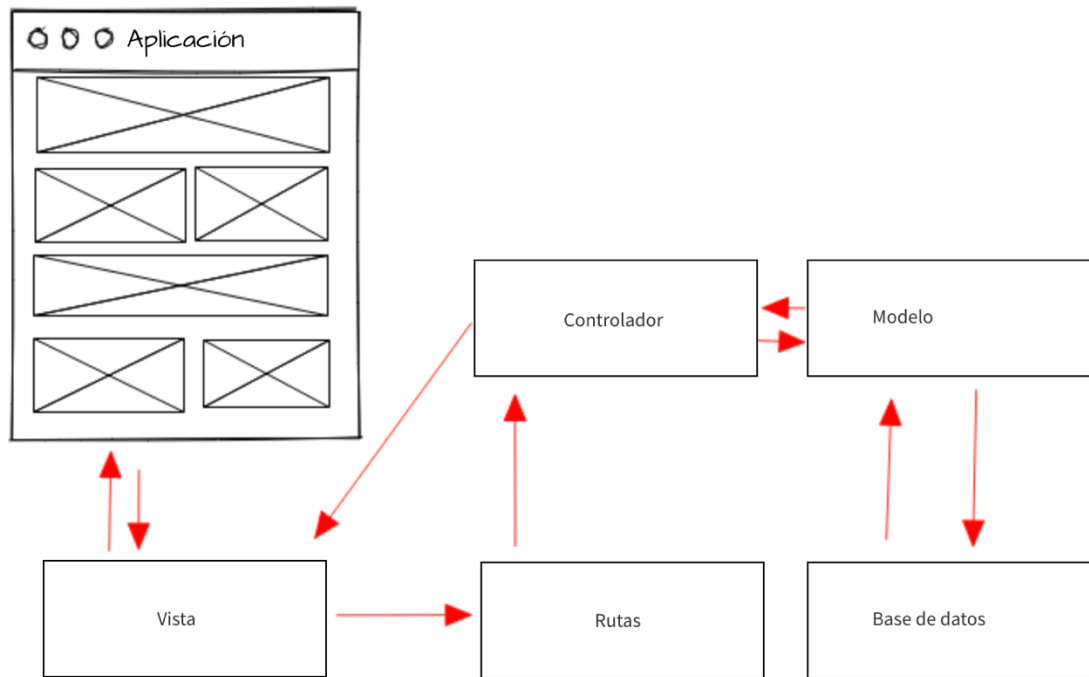
Fuente: Proyecto PIE “Sistema de Información para la Administración y Gestión de Eventos y Escenarios Académicos en el CCAV de Pitalito - AssistApp” – Jhon Perez

En este escenario, los estudiantes realizan peticiones al servidor web, el cual es el encargado de entregar y administrar los recursos y servicios que son usados por el estudiante. Los servicios en este caso serían los ofrecidos por la aplicación móvil almacenados en las tiendas App store (IOS) y Play Store (Android) y se encargarán de enviar las peticiones al web service y procesar las acciones que realicen los estudiantes al hacer uso de la aplicación, enviando una respuesta. El servidor maneja Apache como su servidor web para alojar la aplicación y hace uso del servidor de base de datos MySQL.

Los administradores acceden a los recursos por medio de un navegador web, realizan peticiones al servidor y éste procesa la petición y devuelve una respuesta, como por ejemplo una página HTML.

El siguiente diagrama de componentes muestra el patrón de arquitectura y diseño modelo-vista-controlador de la aplicación:

Ilustración 14. Diagrama de componentes del modelo-vista-controlador de Aplicación móvil



Fuente: Sebastián Gamboa. Wireframe Pro.

El diagrama de componentes anterior muestra el patrón de arquitectura y diseño modelo-vista-controlador. En este caso, el desde la aplicación móvil realiza una petición HTTP a la aplicación web donde hay un Web Service construido en Laravel. Laravel usa lo que llama enrutadores para definir las reglas que rigen cómo se accede a los servicios ofrecidos por la aplicación. Los enrutadores

notifican a los controladores cómo debe responder la aplicación de acuerdo con el tipo de petición que el cliente hace, y luego el controlador se encarga de enviar la respuesta adecuada al cliente.

El controlador si es necesario notificará a uno o varios modelos de que necesita datos alojados en la base de datos, y luego envía estos datos a la vista para ser mostrada en la aplicación del estudiante. La vista es la Aplicación descargada por el estudiante.

3.5.4 Código Fuente

A continuación, se presenta una vista de la aplicación y se mostrará en detalle cómo el código fuente de la aplicación opera para lograr la funcionalidad para la que fue elaborada.

En la siguiente vista se muestra la interfaz de usuarios donde los usuarios registrados pueden iniciar sesión con su correo y número de identificación como cedula y los usuarios que aún no son estudiantes tienen la opción de registrarse.

Para mayor agilidad en este paso los estudiantes pueden iniciar sesión con su correo electrónico y como contraseña predeterminada el sistema le asigna el número de cedula.

Las personas interesadas que aún no se encuentran registradas como estudiantes en el sistema tienen la opción de ingresar datos personales que les brindan acceso a la aplicación y que también son enviados a la universidad para ser procesados.

Los gráficos pueden ser modificados según sea requerido por los administrador y auditores

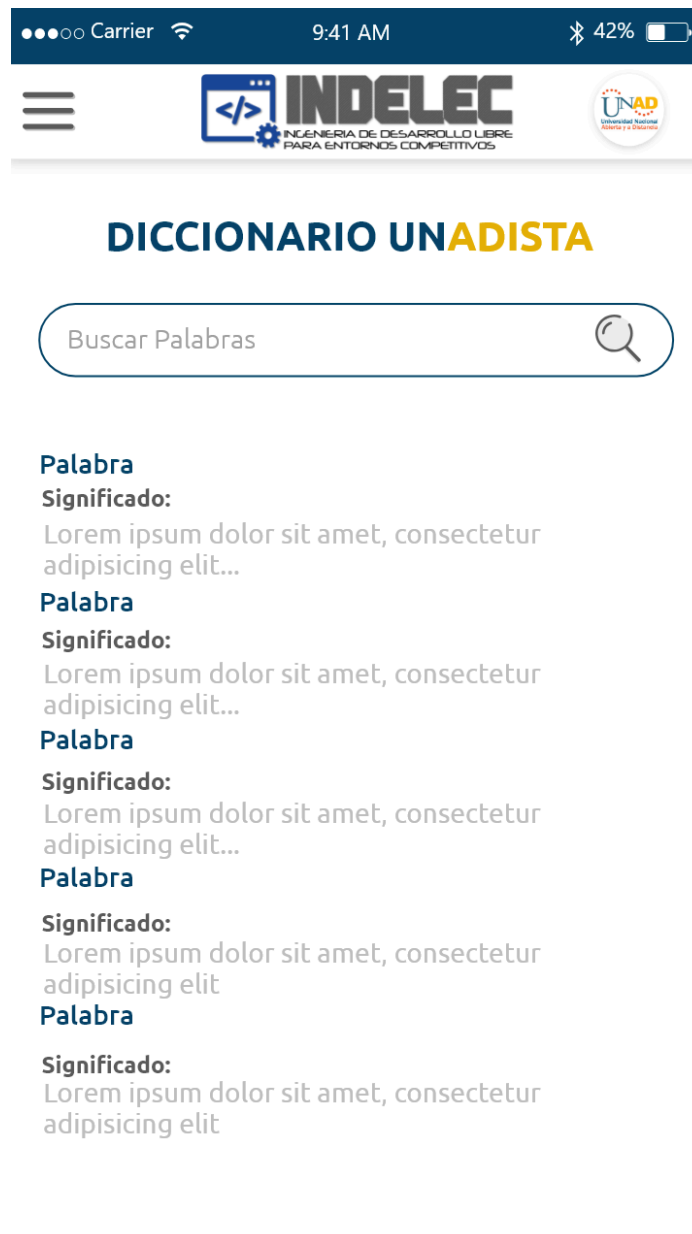
Ilustración 15. Vista login

The illustration shows a mobile app login screen. At the top, there's a status bar with 'Carrier', signal strength, time '9:41 AM', and battery '42%'. The background is a dark, semi-transparent image of a modern building. In the center, there's a circular logo for 'INDELEC' (Instituto Nacional de Desarrollo Electrónico) featuring a blue square with a white code symbol '</>' and a gear. Below the logo, there are two input fields: 'Correo Electronico' with the placeholder 'ejemplocoreroelectronico@gmail.com' and 'Contraseña' with a masked password '.....'. A large, dark blue button with the text 'INICIAR' is positioned below the password field. At the bottom, there are two links: 'Olviaste tu Contraseña ?' and 'REGÍSTRARSE'.

Fuente: Sebastián Gamboa. Adobe xd.

A continuación, se muestra el diccionario Unadista que contiene toda la terminología institucional de la UNAD los estudiantes cuentan con un buscador avanzado para realizar consultas.

Ilustración 16. Vista diccionario Unadista



Fuente: Sebastián Gamboa. Adobe xd.

La aplicación cuenta con un menú con las opciones básicas para navegar entre las secciones de la misma

Ilustración 17. Menu Aplicación



Fuente: Sebastián Gamboa. Adobe xd.

La aplicación muestra lo eventos a realizarse en el centro más cercano gracias a su módulo de geolocalización. La aplicación toma la localización actual y de acuerdo a ella le muestra información sobre el centro más cercano.

Ilustración 18. Geolocalización y eventos.

| Escuela | Evento | Lugar | Hora |
|---------|---|----------|-----------------------------|
| ECBTI | ACADEMICO / B-Learning Herramientas digitales para la gestión del conocimiento. | Aula 103 | 08:00:00 hasta las 10:00:00 |
| ECSAH | ACADEMICO / componente practico escenario 2 | Aula 110 | 08:00:00 hasta las 10:00:00 |

Fuente: Sebastián Gamboa. Adobe xd.

El código fuente detrás del funcionamiento de los procesos que se acaba de mostrar se explicará a continuación. Primero existe una fase ya desarrollada que

corresponde a un servicio web donde se administra y gestiona los eventos de la Universidad. Para la interacción entre estos módulos web y la aplicación móvil se creó un web service hecho en Laravel que contendrá una API que enviará datos a la Aplicación.

Ilustración 19. Packages y plugins usados para el desarrollo

```
{
  "name": "AssitApp",
  "version": "0.0.1",
  "private": true,
  "scripts": {
    "start": "node node_modules/react-native/local-cli/cli.js start",
    "test": "jest"
  },
  "rnpm": {
    "assets": [
      "./public/fonts/"
    ]
  },
  "dependencies": {
    "numeral": "^2.0.6",
    "react": "16.3.0-alpha.3",
    "react-native": "0.54.3",
    "react-native-app-link": "^1.0.0",
    "react-native-cache-store": "^1.0.2",
    "react-native-elements": "^0.19.1",
    "react-native-hide-show-password-input": "^1.0.7",
    "react-native-keyboard-spacer": "^0.4.1",
    "react-native-numeric-input": "^1.3.1",
    "react-native-picker-select": "^4.4.0",
    "react-native-simple-radio-button": "^2.7.2",
    "react-native-table-component": "^1.1.8",
    "react-native-vector-icons": "^4.6.0",
    "react-native-video": "^2.2.0",
    "react-native-video-player": "^0.9.1",
    "react-navigation": "^2.2.5"
  },
  "devDependencies": {
    "babel-jest": "22.4.3",
    "babel-preset-react-native": "4.0.0",
    "jest": "22.4.3",
    "react-test-renderer": "16.3.0-alpha.3"
  },
  "jest": {
    "preset": "react-native"
  }
}
```

Fuente: Sebastián Gamboa. Php storm.

En la ilustración podemos ver el archivo Package.json que contiene todas las herramientas usadas para el desarrollo de la aplicación tenemos que mi versión de React Native usada es la 0.54.3 que es la versión más estable que existe en este momento también encontramos el nombre de nuestra App, el número de versión y todos los paquetes usados en la etapa de desarrollo

Ilustración 20. Archivo de rutas de la Aplicación

```
import React from "react";
import {createStackNavigator,DrawerActions} from 'react-navigation';
import {
  TouchableOpacity,
  Image,
  Dimensions,
  Platform,
  Button
} from "react-native";

import DrawerScreen from './menu/DrawerScreen';
import Load from './screens/Load';
import CacheStore from 'react-native-cache-store';
import Points from './screens/Points';
import OffConnection from './screens/OffConnection';
import CategorieList from './screens/CategorieList';

const {width} = Dimensions.get('window');

let img = null;
setInterval(()=>{
  CacheStore.get('image').then((value) => {
    if(value === 1 ){
      img = require('../public/images/carShopRed.png');
    }else{
      img = require('../public/images/carShop.png');
    }
  });
},100);

export const HomeStack = createStackNavigator({
  screen_load:{
    screen:Load,
    navigationOptions:{
      header:null,
      drawerLockMode:'unlocked',
    }
  },
  screen_Off:{
    screen:OffConnection,
    navigationOptions:{
      header:null,
      drawerLockMode:'unlocked',
    }
  }
},
```

Fuente: Sebastián Gamboa. Php storm.

En esta imagen se muestra el archivo de rutas de la Aplicación y cada uno de sus atributos por ejemplo en `screen_load` le decimos que no muestre ningún header y con el atributo `drawerLockMode` le decimos que no muestre la caja de menú en la precarga de la aplicación

Ilustración 21. Conexión a laravel

```
const URI = 'http://159.89.233.208';

export default {
  async version() {
    try {
      let response = await fetch(URI + '/api/version');
      let responseJsonData = await response.json();
      return responseJsonData;
    } catch (e) {
      console.log(e)
    }
  },

  async categories() {
    try {
      let response = await fetch(URI + '/api/categories');
      let responseJsonData = await response.json();
      return responseJsonData;
    } catch (e) {
      console.log(e)
    }
  },

  async orders(id) {
    try {
      let response = await fetch(URI + '/api/pedidos/' + id);
      let responseJsonData = await response.json();
      return responseJsonData;
    } catch (e) {
      console.log(e)
    }
  },

  async category(id) {
    try {
      let response = await fetch(URI + '/api/react/' + id);
      let responseJsonData = await response.json();
      return responseJsonData;
    } catch (e) {
      console.log(e)
    }
  },

  async searchProd(prod) {
    try {
      let response = await fetch(URI + '/api/search/' + prod);
      let responseJsonData = await response.json();
      return responseJsonData;
    } catch (e) {
      console.log(e)
    }
  },
}
```

Fuente: Sebastián Gamboa. Php storm.

En la ilustración anterior se muestra el archivo de conexión al web service hecho en laravel, cada función representa una consulta hecha desde la aplicación que se envía a un api que está en el servidor programada con el framework laravel. Recordemos que las bases de datos de esta aplicación ya se encontraban echas en su gran mayoría en la primera fase lo que se hizo fue crear un web service y conectarlo a la base de datos que ya existía y mediante una Api diseñada en sincronía con la aplicación y la necesidad de datos que surgían conforme se avanzaba

Ilustración 22. Modelado de ingreso de texto en React Native

```
<TextInput
  style={styles.inputData}
  placeholder={'nombre'}
  ref='nombre'
  underlineColorAndroid={"transparent"}
  autoComplete={false}
  keyboardAppearance={'dark'}
  keyboardType={'default'}
  maxLength={30}
  returnKeyType={'next'}
  onChangeText={(data) => this.setState({nombre: data})}
  onSubmitEditing={() => {
    this.refs.apellido.focus();
  }}
/>
```

Fuente: Sebastián Gamboa. Php storm.

En el código anterior se muestra un ejemplo de las entradas de texto que se usaron en toda la aplicación. Por ejemplo este en un campo de ingreso de texto que el HTML se conoce como input de tipo texto en React Native contamos con una serie de atributos muy amplia entre los que están el tipo y color del teclado a mostrar en el celular cuando un input tiene el foco, se puede elegir si desea usar opciones como autocompletado o autocorrección de ortografía y al igual que en HTML cuando se ejecuta un evento puedes enviar algún tipo de acción o función.

CAPITULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- Fue de gran ayuda la participación de las personas encargadas de registro y control el CCAV Pitalito como la ayuda y sugerencias que aportaron los usuarios y administradores de la parte web que se desarrolló en la primera fase de este proyecto. En este proceso se auditaron diferentes áreas y actividades que realizan tutores y estudiantes con el fin de encontrar la necesidad y mejorar los requerimientos de la aplicación. Las siguientes personas fueron entrevistadas con tal fin:

Tabla 4. Casos de uso diccionario Unadista

| Oficina | Funcionario | Cargo |
|--------------------|-----------------|--------------------|
| Registro y control | | Registro y control |
| Registro y control | | Registro y control |
| Estudiante | Natalia Medina | Estudiante |
| Estudiante | Fernando Claros | Estudiante |
| Tutor | Mireya Gómez | Tutor |

Fuente: autor del proyecto

Los problemas que se identificaron son los siguientes: El más común es que los canales de comunicación actuales que usa la UNAD para conectar con sus estudiantes, cuando de eventos o reuniones se trata no es tan efectivo y por lo general cuando los estudiantes ven los mensajes o correos electrónicos muchas veces ya es demasiado tarde para participar los estudiantes manifiestan que no participan en los eventos de la universidad porque muchas veces ni siquiera se enteran. Por parte de los tutores se manifiesta la misma necesidad la falta de comunicación entre tutor y estudiante cuando se trata de eventos o algún tipo de

convocatoria que sea diferente a lo programado en la plataforma de la UNAD. Las personas de registro y control nos manifiestan la necesidad de un mecanismo de control para ellos llevar control de los eventos para así asignar aulas y los elementos necesarios para su desarrollo.

Desarrollar esta aplicación requirió investigar y aprender muchas tecnologías del lado web como Framework Laravel, JavaScript, css, html, php. Manejo de base de datos Mysql y en la parte móvil React Native. Estudiando a fondo y haciendo muchos ejercicios se adquieren conocimientos que seguro serán de mucha utilidad en el futuro y que sirven como complemento a lo ya aprendido durante el transcurso de la carrera de ingeniería de sistemas. Del lado del servidor fue de gran ayuda el Framework Laravel ya que gracias a su estructura modelo – vista – controlador se pudo implementar de manera limpia y segura el api que necesitaba para conectar con la aplicación móvil. Dado el conjunto de funcionalidades que posee la tecnología Laravel, se agilizaron tareas que son comunes en el escenario de desarrollo de la aplicación web, tales como el procesamiento de datos, el control de rutas, y elementos básicos de seguridad, ahorrando y acortando tiempo de codificación. Por otro lado, Laravel permitió trabajar fácilmente las operaciones CRUD en la base de datos, por medio de su componente Eloquent, y el motor de plantillas llamado Blade, que facilitó la generación de las vistas a mostrar en el lado del cliente, es decir, las páginas que componen la aplicación.

- Se desarrolló una aplicación móvil siguiendo la metodología de desarrollo SCRUM, por medio de la cual se definieron los ciclos temporales, funcionales de la aplicación, planeación de la iteración, se analizaron los requerimientos, se realizó un diseño de la aplicación y posteriormente se realizó la inspección y adaptación. Esta metodología es un conjunto de buenas prácticas para trabajar y obtener el mejor resultado posible en SCRUM se realiza entregas parciales y regulares del producto final priorizadas por el beneficio que aportan al receptor del proyecto.

En el desarrollo de esta aplicación se implementaron 3 módulos. Dos módulos en la aplicación móvil y otro del lado del servidor. Del lado del servidor creamos web service encargado de la conexión y consulta a base de datos hecho en Laravel. Por parte de la aplicación móvil se crean dos módulos uno para el diccionario Unadista donde los estudiantes usuarios de la aplicación puede consultar términos y siglas institucionales y un segundo módulo de inscripción a

eventos donde los estudiantes pueden consultar los eventos que se realizaran en la universidad y podrán inscribirse para participar en los mismos.

4.2 RECOMENDACIONES

Se recomienda que de manera constante se esté alimentando la base de datos de la aplicación en el área de diccionario de esto dependerá el éxito y funcionalidad de la aplicación para que los estudiantes la encuentren útil y que al momento de hacer consultas referentes a la universidad la aplicación esta lo suficientemente completa para responder todo requerimiento. Del lado de las citas se recomienda el uso contante y obligatorio por parte de los tutores que todos los eventos que se realicen en la universidad se carguen por medio del aplicativo web diseñado en la primera fase al que ya tienen acceso los tutores del CCAV Pitalito. La aplicación se diseñó de manera integral para que se puedan adicionar módulos o nuevas funciones a futuro.

Se recomienda que la aplicación sea alojada en un servidor VPS que cuente con al menos 8 GB de RAM, un almacenamiento de 1 TB y con un poder de procesamiento de cuatro núcleos, para tratar de minimizar problemas de acceso por parte de usuarios, y problemas de rendimiento y almacenamiento de la aplicación web y el servidor. Es necesario y recomendable que el servidor tenga instalado una versión de PHP 7.0 ya que en esta versión se desarrolló la aplicación. Y por parte de la aplicación móvil se recomienda el pago anual de los servicios de almacenamiento en Play Store (Android) y App Store (IOS) para que la aplicación siempre esté disponible para los estudiantes.

CAPITULO V. BIBLIOGRAFÍA

GILBERT GINESTÁ, M., & PEÑA GONZÁLEZ, Á. (2005). Ingeniería del Software en Entornos SL. Barcelona: Fundación para la Universitat Oberta de Catalunya.

JAMES H. STEPHENS, J. (31 de 07 de 2001). *patents.google.com*. Obtenido de <https://patents.google.com/patent/US6633324B2/en>

MORA, S. L. (2002). Programación de aplicaciones web: historia, principios básicos y clientes web. En S. L. Mora, Programación de aplicaciones web: historia, principios básicos y clientes web.

PIRZADA, S. L. (17 de 09 de 2004). *patents.google.com*. Obtenido de *patents.google.com/patent/US7693734B2/en*

PRESSMAN, R. S. (2002). El Modelo Espiral. En R. S. Pressman, Ingeniería del Software un Enfoque Práctico Quinta Edición (págs. 24-26). Madrid: Mc Graw Hill

AVELLA MARTINEZ, Laura Yaneth y PARRA RUIZ, Paola Patricia. Tecnologías de la información y la comunicación (TICS) en el sector salud. [En línea]. 2013. Disponible en internet: < <http://bdigital.unal.edu.co/11172/1/laurayanethavellamartinez.2013.pdf> >

DOVZHENKO, Oleksii. Why Laravel will become most popular PHP MVC framework in 2018 [En línea]. LINKEDIN, 2018.[citado 25 de julio de 2018] Disponible en internet: < <https://www.linkedin.com/pulse/why-laravel-become-most-popular-php-mvc-framework-2018-dovzhenko> >

ECURED. Aplicación web [En línea]. Cuba. Oficina nacional para la informatización. s.f. rev. 13 de septiembre de 2013. [citado mayo 02 de 2018] Disponible en internet: < https://www.ecured.cu/Aplicaci%C3%B3n_web >

GOBIERNO DE COLOMBIA. Plan Nacional de Desarrollo. [En línea]. Colombia. Departamento nacional de planeación. 2010. [Citado 14 de marzo de

2018]. Disponible en internet: <<https://www.dnp.gov.co/Plan-Nacional-de-Desarrollo/PND-2010-2014/Paginas/Plan-Nacional-De-2010-2014.aspx>>

LUCID SOFTWARE INC. ¿Qué es el lenguaje unificado de modelado (UML)? [En línea]. [Citado mayo 02 de 2018] Disponible en internet: <<https://www.lucidchart.com/pages/es/qu%C3%A9-es-el-lenguaje-unificado-de-modelado-uml>>

MARTINIG & ASSOCIATES. Agile Development with ICONIX process [En línea]. [Citado mayo 2 de 2018]. Disponible en internet: <http://www.methodsandtools.com/archive/archive.php?id=22>

MOSTAFA, Ashour A. N. A review of different agile methods. [En línea]. 2016. Disponible en internet: <https://www.researchgate.net/publication/301346660_Five_different_agile_methods>

MOZILLA DEVELOPER NETWORK. ¿Qué es un servidor WEB? [En línea]. [Citado 2 de mayo de 2018]. Disponible en internet: https://developer.mozilla.org/es/docs/Learn/Common_questions/Que_es_un_servidor_WEB

PRESSMAN, Roger S. Ingeniería de software. Un enfoque práctico. 7 ed. México: McGraw-Hill, 2010, 12 p.

TAYLOR OTWELL. Introduction [En línea]. [Citado 2 de mayo de 2018]. Disponible en internet: <https://laravel.com/docs/4.2/introduction>

TECHTARGET. Database (DB) [En línea]. [Citado 3 de mayo de 2018]. Disponible en internet: <<https://searchsqlserver.techtarget.com/definition/database>>

THE COMPUTER TECHNOLOGY DOCUMENTATION PROJECT. UML expanded use case. [En línea]. [Citado 15 de mayo de 2018]. Disponible en internet: <<http://www.comptechdoc.org/independent/uml/begin/umlexusecase.html>>

THE PHP GROUP. What is PHP? [En línea]. [Citado 2 de mayo de 2018]. Disponible en internet: <<http://php.net/manual/en/intro-what-is.php>>

UNIDAD CENTRAL DEL VALLE DEL CAUCA. Reseña histórica [En línea]. Colombia. s.f. rev. 5 de junio de 2018. [Citado 6 de junio de 2018]. Disponible en

internet: < <http://www.uceva.edu.co/index.php/institucional/nosotros/resena-historica-uceva> >

UNIDAD CENTRAL DEL VALLE DEL CAUCA. Direccionamiento estratégico. [En línea]. Colombia. s.f. rev. 5 de junio de 2018. [Citado 6 de junio de 2018]. Disponible en internet: < <http://www.uceva.edu.co/index.php/institucional/nosotros/direccionamiento-estrategico-uceva> >

UNIDAD CENTRAL DEL VALLE DEL CAUCA. Direccionamiento estratégico. [En línea]. Colombia. s.f. rev. 5 de junio de 2018. [Citado 6 de junio de 2018]. Disponible en internet: < <http://www.uceva.edu.co/index.php/institucional/nosotros/direccionamiento-estrategico-uceva> >

UNIDAD CENTRAL DEL VALLE DEL CAUCA. Programa de salud. [En línea]. Colombia. s.f. rev. 5 de junio de 2018. [Citado 6 de junio de 2018]. Disponible en internet: < <http://www.uceva.edu.co/index.php/bienestar-programa-de-salud> >

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA. Diagramas de casos de uso. [En línea]. Colombia. [Citado 15 de mayo de 2018]. Disponible en internet: < http://stadium.unad.edu.co/ovas/10596_9839/diagramas_de_casos_de_uso.html >

VISUAL PARADIGM. Sequence diagram [En línea]. [Citado 15 de mayo de 2018]. Disponible en internet: < <https://www.visual-paradigm.com/learning/handbooks/software-design-handbook/sequence-diagram.jsp> >